

IIP-113-A

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: Kin
Serial Number: Unknown
Filed: Concurrently herewith
Group Art Unit: Unknown
Examiner: Unknown
Confirmation Number: Unknown
Title: VEHICLE SPEED MEASURING APPARTUS

TRANSMITTAL OF PRIORITY DOCUMENT

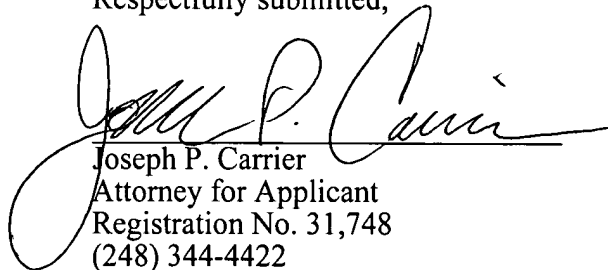
Commissioner For Patents
PO Box 1450
Alexandria, VA 22313-1450

Sir:

In connection with the identified application, applicant encloses for filing a certified copy of:
Japanese Patent Application No. 2002-308302, filed 23 October 2002, to support applicant's claim for
Convention priority under 35 USC §119.


Respectfully submitted,

Customer Number 21828
Carrier, Blackman & Associates, P.C.
24101 Novi Road, Suite 100
Novi, Michigan 48375
23 October 2003


Joseph P. Carrier
Attorney for Applicant
Registration No. 31,748
(248) 344-4422

I hereby certify that this correspondence is being deposited with the U.S. Postal Service as Express
Mail Certificate ET986050195US in an envelope addressed to Mail Stop Patent Application,
Commissioner For Patents, PO Box 1450, Alexandria VA 22313-1450 on 23 October 2003.

Dated: 23 October 2003
JPC/km
enclosures


Kathryn MacKenzie

日 本 国 特 許 庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日
Date of Application:

2002年10月23日

出 願 番 号
Application Number:

特願2002-308302

[ST.10/C]:

[JP 2002-308302]

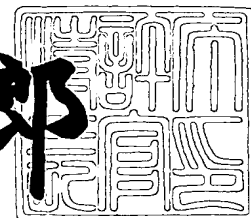
出 願 人
Applicant(s):

本田技研工業株式会社

2003年 6月 2日

特 許 庁 長 官
Commissioner,
Japan Patent Office

太田信一郎



出証番号 出証特2003-3042342

【書類名】 特許願

【整理番号】 H102272001

【提出日】 平成14年10月23日

【あて先】 特許庁長官 殿

【国際特許分類】 B60C 23/06
B60T 8/58

【発明者】

【住所又は居所】 埼玉県和光市中央1丁目4番1号
株式会社本田技術研究所内

【氏名】 金 圭勇

【特許出願人】

【識別番号】 000005326

【氏名又は名称】 本田技研工業株式会社

【代理人】

【識別番号】 100064414

【弁理士】

【氏名又は名称】 磯野 道造

【電話番号】 03-5211-2488

【手数料の表示】

【予納台帳番号】 015392

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9713945

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 車体速測定装置

【特許請求の範囲】

【請求項 1】

少なくとも入力部と処理部を有し、

タイヤを介して入力される路面との振動を検知する前輪側及び後輪側の振動検出センサから検出値をそれぞれ入力し、この入力した検出値の変化のパターンに基づいて車両の車体速を測定する車体速測定装置であって、

前記入力部から前記検出値を入力すると、前記処理部にて、前記前輪側及び後輪側について前記検出値の変化のパターンをタイヤ固有の影響を除去してそれぞれ特徴抽出し、この特徴抽出した検出値の変化のパターンを前記前輪側と後輪側とでパターンマッチングし、一致したパターンの時間差を求め、このように求めた時間差と予め記憶している基準長さとかから車体速を演算するようにした構成を有すること、

を特徴とする車体速測定装置。

【請求項 2】

前記振動検出センサが車輪速センサであること、を特徴とする請求項 1 に記載の車体速測定装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、車両の車体速を測定する車体速測定装置に関する。

【0002】

【従来の技術】

車両における各種制御に車体速が用いられるが、この車体速は一般的には車輪速センサの検出値を用いて演算等により測定される。例えば、制動時のブレーキロックを防止する装置や車両挙動制御装等では、車輪速センサの検出値から演算により、車体速を測定していた。例えば、各車輪速センサから得られる4輪の車輪速のうち、一番早いものを車体速にしたり、従動輪の車輪速の平均を取って車

体速にしたりしている（例えば特許文献 1 等参照）。

【 0 0 0 3 】

【特許文献 1】

特開 2 0 0 1 - 4 7 9 9 8 号公報（例えば、請求項 2、3、段落 0 0 3 1、0 0 3 2 等、図 5 等）

【 0 0 0 4 】

【発明が解決しようとする課題】

しかしながら、車輪速センサは、タイヤ（車輪）がある時間内に何回回転したかにより車輪速を演算するものである。このため、異径のタイヤを装着した場合やタイヤの空気圧が変化した場合にはタイヤ径そのものが変化するもので、車輪速もタイヤ径により変化してしまう。よって、車輪速に基づいて演算により測定される車体速も変化してしまう。この問題は、変速機の出力側シャフトの回転速度を検出することで車速を測定する車速センサの場合にもいえることであり、タイヤ径が変化すると車速（車体速）も変化してしまう。

【 0 0 0 5 】

そこで、本発明は、タイヤ径の変化に影響されない車体速の測定を可能とした車体速測定装置を提供することを主たる目的とする。

【 0 0 0 6 】

【課題を解決するための手段】

前記課題に鑑み、本発明者は鋭意研究を行い、振動検出センサ（車輪速センサ）の検出値は、路面の段差やバンプ（路面の凸凹）等により変動すること。この変動は、車両が前進している場合は、まず前輪のセンサの検出値に現われ、次に後輪のセンサの検出値に現われること。ここで、前輪における検出値の変動と、後輪における検出値の変動が同じ段差やバンプが原因であることが判り、かつ両変動の時間間隔が判れば、車両のホイールベース間距離（基準長さ、前方と後方のセンサ間距離）から車体速を測定することができること、つまりタイヤ径の変化に影響されない車体速の測定が可能であること等に着目し、本発明を完成するに至った。

【 0 0 0 7 】

即ち、前記課題を解決した本発明は、少なくとも入力部と処理部を有し、タイヤを介して入力される路面との振動を検知する前輪側及び後輪側の振動検出センサから検出値をそれぞれ入力し、この入力した検出値の変化のパターンに基づいて車両の車体速を測定する車体速測定装置である。この車体速測定装置は、前記入力部から前記検出値を入力すると、前記処理部にて、前記前輪側及び後輪側について前記検出値の変化のパターンをタイヤ固有の影響を除去してそれぞれ特徴抽出し、この特徴抽出した検出値の変化のパターンを前記前輪側と後輪側とでパターンマッチングし、一致したパターンの時間差を求め、このように求めた時間差と予め記憶している基準長さとかから車体速を演算するようにした構成を有することを特徴とする。

【 0 0 0 8 】

この構成は、道路上のある地点（ある任意のバンプ等）を、ある長さの物体（車両）がどの程度の時間をかけて通過したかにより車体速を測定するものであり、従来技術のようにタイヤがある時間内に何回回転したかにより車体速を測定するものとは原理的に異なる。このため、タイヤ径の変化に影響されない（本質的に影響されない）車体速を測定することができる。

ちなみに、後記する実施形態では、前輪側で出現した車輪速の検出値の変動と原因を同じくする変動が、いつ後輪側で起こったかをパターンマッチングにより検出することとして車体速を測定する実施形態を説明しているが、逆に、後輪側で出現した車輪速の検出値の変動と原因を同じくする変動が、遡ることどのくらい前に前輪側で起こったのかをパターンマッチングにより検出することとして車体速を測定する実施形態としてもよい。

【 0 0 0 9 】

また、本発明は、請求項 1 の構成において、前記振動検出センサが車輪速センサであることを特徴とする。

車輪速センサは路面のバンプ等により検出値が変動する。即ち、車輪速センサは、振動検出センサとして利用することができる。また、車輪速センサは一般的なセンサであり、多くの車両に搭載されていることから、別に車輪速センサを準備する必要がない。

【 0 0 1 0 】

【発明の実施の形態】

以下、本発明の本実施形態を、図面を参照して詳細に説明する。

参照する図において、図 1 は本実施形態の車体速測定装置を搭載する車両のシステム構成図である。

【 0 0 1 1 】

《車体速測定装置の構成》

本実施形態の車体速測定装置 1 の構成を、車両 C の構成を含めて説明する。

図 1 に示すように、本実施形態の車体速測定装置 1 を搭載する車両 C は、前輪 W_f 2 つ、後輪 W_r 2 つを有する 4 輪車両である。この車両 C は、右側の前輪 W_f 及び右側の後輪 W_r に、請求項における「振動検出センサ」に相当する車輪速センサ V_S (V_{Sf} , V_{Sr}) を有する。なお、本明細書において、符号に添えられる f , r の添字は、 f が前輪側、 r が後輪側であることを示す。

【 0 0 1 2 】

車輪速センサ V_S (V_{Sf} , V_{Sr}) は、例えばホール素子等を用いて車速パルスを生成する一般的なセンサである。この車輪速センサ V_{Sf} , V_{Sr} が生成し、車体速測定装置 1 に送信する車速パルス（アナログ電気信号）は、車速が早くなるほど単位時間当たりのパルス数が多くなり、車速が遅くなるほど単位時間当たりのパルス数が少なくなる。一般的には、この車速パルスに基づいて車輪速や車体速を測定するが、既に説明したように、この車速パルスから得られる車輪速や車体速は、タイヤの空気圧の影響、装着するタイヤの種類（外径サイズ）等により変化し、必ずしも正確なものではない。なお、ブレーキロックを防止するシステムを搭載する車両やトラクションを制御するシステムを搭載した車両は、通常、車輪速センサ V_S を有しているので、これを流用することができる。

【 0 0 1 3 】

次に、車体速測定装置 1 は、図示しないマイコン（マイクロコンピュータ）及び周辺回路から構成され、マイコンが図示しない ROM に書き込まれたプログラムを読み出すことにより該プログラムの各モジュール（後記するデジタルフィルタ 12、正規化手段 15 等）を実行して、車体速の測定を行う。また、車体速測

定装置 1 は、車体速の測定を行うために、各種信号・情報・指令等を入出力する入出力ポート（後記する入出力インタフェイス 1 1）、アナログ信号をデジタル信号に変換してマイコンでデジタル処理するための図示しない A D 変換器等を有する。なお、入出力ポートは請求項の「入力部」に相当する。

【 0 0 1 4 】

車体速測定装置 1 について、図 2、図 3 を参照してさらに詳細に説明する。図 2 は、車体速測定装置のブロック構成図である。図 3 は、車輪速の検出値の変動を説明する図である。

【 0 0 1 5 】

図 2 に示すように、車体速測定装置 1 は、主に、入出力インタフェイス 1 1、デジタルフィルタ 1 2（1 2 f, 1 2 r）、バッファコントローラ 1 3（1 3 f, 1 3 r）、データバッファ 1 4（1 4 f, 1 4 r）、正規化手段 1 5（1 5 f, 1 5 r）、相互相関関数演算手段 1 6、最大値抽出手段 1 7、車体速演算手段 1 8、平均車体速演算手段 1 9 を含んで構成される。

【 0 0 1 6 】

入出力インタフェイス 1 1 は、車体速測定装置 1 で処理するデータの入力及び車体速測定装置 1 で処理したデータの出力を行う機能を有する。なお、車体速測定装置 1 では、車速パルスは、デジタルデータの車輪速（検出値 V （ V_f , V_r ））として取り扱うものとする。ちなみに、本実施形態での車輪速のサンプリングレートは 1 0 0 0 H z とする。

【 0 0 1 7 】

デジタルフィルタ 1 2（1 2 f, 1 2 r）は、刻々と入力される車輪速の検出値 V （ V_f , V_r ）を処理して、特定の周波数の成分のみを通過させるデジタル式のバンドパスフィルタである。このように特定の周波数のみを通過させるのは、タイヤのユニフォーミティの崩れによる車輪速の変動を除去して、路面のバンブや段差等（以下「路面バンブ等」という）による車輪速変動を抽出するためである。

【 0 0 1 8 】

つまり、タイヤはゴムやスチールワイヤ等を巻いて製造するため、タイヤ一周

の強度や密度に不均一性（ユニフォーミティの崩れ）が存在する。このため、図 3（a）に示すように車輪 W が路面上を回転すると、見かけ上車両 C が一定速度で走行していても、車輪速センサ V S から得られる検出値 V の時間変動（車輪速検出値の変動曲線）には、図 3（b）に示すように、タイヤのアンバランス（ユニフォーミティの崩れ）による周期の大きな変動が生じる。そして、この周期の大きな変動に、路面バンプ等による周期の小さな変動が重畳される。本実施形態では、路面バンプ等による車輪速変動から絶対的な車体速を求めるものであることから、図 3（c）に示すように、タイヤのユニフォーミティの崩れによる変動成分を、デジタルフィルタ 1 2 で除去し（つまりタイヤ固有の影響を除去し）、後の処理を円滑に行えるようにする。なお、車輪速が早いほど、タイヤのユニフォーミティの崩れによる車輪速変動の周期（周波数）、路面バンプ等による車輪速変動の周期（周波数）、は全体的に短周期になる（高周波数帯域にシフトする）。このため、デジタルフィルタ 1 2 は、車輪速が早くなるほど高い周波数帯域の車輪速の変動を通過するように、車輪速応動に構成してある。

【 0 0 1 9 】

バッファコントローラ 1 3（1 3 f，1 3 r）は、デジタルフィルタ 1 2 を通過した車輪速の検出値 V（V f，V r）を、1 0 ミリ秒間隔を置いて取得し、これをデータバッファ 1 4（1 4 f，1 4 r）に、所定個数書き込む機能を有すると共に、書き込んだ検出値 V を、所定個数まとめて読み出す機能を有する手段である。

【 0 0 2 0 】

データバッファ 1 4（1 4 f，1 4 r）は、検出値 V（V f，V r）を一時記憶する読み書き自在のメモリである。データの読み書きは、バッファコントローラ 1 3 を介して行われる。

【 0 0 2 1 】

ちなみに、検出値 V（V f，V r）は、処理の回数を数える処理カウンタ n，m と対応付けてデータバッファ 1 4 に記憶される。具体的には、前輪側の検出値 V f は、配列変数 V f（n）として処理カウンタ n と対応付けてデータバッファ 1 4 f に記憶される。また、後輪側の検出値 V r は、配列変数 V r（m）として

処理カウンタ m と対応付けてデータバッファ 1 4 r に記憶される。ちなみに、処理カウンタ n , m は、検出値 V (V_f , V_r) を 1 個記憶する際に 1 ずつインクリメントされるカウンタ (インデックス) であり、本実施形態では、処理カウンタ n は 1 6 を上限に 1 ずつインクリメントされ、処理カウンタ m は 3 0 を上限に 1 ずつインクリメントされる。

【 0 0 2 2 】

つまり、前記した所定個数 (終値 N , M) は、前輪側のバッファコントローラ 1 3 f についていえば 1 6 個 (終値 $N = 1 6$) であり、後輪側のバッファコントローラ 1 3 r についていえば 3 0 個 (終値 $M = 3 0$) である。このように、記憶するデータ数を絞り込むのは、後段の正規化手段 1 5 や相互相関関数演算手段 1 6 における演算処理の負荷を少なくするためである。また、このようにデータ数を絞り込んでも、十分に絶対的な車体速 V_v を測定することができるからである。なお、処理カウンタ n , m の初期値はそれぞれ 0 であるが、実際にデータが記憶されるのは 1 からである。従って、処理カウンタ n は実質上 1 ~ 1 6 までの正の整数値を取り、処理カウンタ m は実質上 1 ~ 3 0 までの正の整数値を取る。このように、後輪側の処理カウンタ m の終値 M が大きな値を取るのは、前輪側で起こったのと同じ事象 (特定のバンプ等を通じたことによる検出値 V の変動) は時間を置いて後輪側で起こるが、後輪側で起こった際にその事象を見逃さないためである。よって、終値 M は、前輪側で起こったのと同じ事象を確実に記憶できる数が設定される。

【 0 0 2 3 】

ちなみに、本実施形態では、データバッファ 1 4 がデジタルフィルタ 1 2 から検出値 V を取得する間隔が 1 0 ミリ秒置きであることから、処理カウンタ n が 1 6 になるまで検出値 V_f を配列変数 $V_f(n)$ に格納すると、検出値 V_f を実時間にして 1 5 0 ミリ秒分データバッファ 1 4 f に記憶したことになる (1 5 0 ミリ秒 = $(1 6 - 1) \times 1 0$ ミリ秒)。同様に、処理カウンタ m が終値の 3 0 になるまで検出値 V_r を配列変数 $V_r(m)$ に格納すると、検出値 V_r を実時間で 2 9 0 ミリ秒分データバッファ 1 4 r に記憶したことになる (2 9 0 ミリ秒 = $(3 0 - 1) \times 1 0$ ミリ秒)。

【 0 0 2 4 】

なお、車速（車体速、車輪速）が早い場合は、データの取得間隔（サンプリング間隔）を短くするのが好ましい。その一方で、さほど長い時間分検出値 V を記憶する必要はない。逆に、車速が遅い場合は、長い時間分検出値 V を記憶する必要がある（つまり、終値 M を大きくするか、サンプリング間隔を長くするかが必要）。

【 0 0 2 5 】

次に、正規化手段 1 5（1 5 f，1 5 r）を説明する。

前輪側の正規化手段 1 5 f は、バッファコントローラ 1 3 f を介してデータバッファ 1 4 f から配列変数 $V f(n)$ を 1 6 個分、全てを読み出す機能を有する。そして、次の相互相関関数演算手段 1 6 での処理を行い易くするため、検出値 $V f$ （＝配列変数 $V f(n)$ ）から車輪速成分を取り除いて正規化する機能を有する。このため、正規化手段 1 5 f は、配列変数 $V f(1) \sim V f(16)$ までの平均車輪速 $A V f$ を求める処理を行う。なお、前輪側の平均車輪速 $A V f$ は、次の式 1 で演算される。

$$\begin{aligned} A V f &= \Sigma V f(n) / 16 \\ &= (V f(1) + V f(2) + \dots + V f(16)) / 16 \quad \dots \quad (\text{式 1}) \end{aligned}$$

【 0 0 2 6 】

また、正規化手段 1 5 f は、配列変数 $V f(n)$ の正規化を次の式 2 のように行い、車輪速成分（平均車輪速 $A V f$ ）を取り除く。

$$V f(n) = V f(n) - A V f \quad \dots \quad (\text{式 2})$$

【 0 0 2 7 】

なお、処理カウンタ n （データを記憶している部分）は 1 ～ 1 6 までの正の整数であるので、正規化手段 1 5 f は、処理カウンタ n を 1 から 1 ずつインクリメントして終値 N の 1 6 になるまで 1 6 回、式 2 を実行する。これにより、正規化した配列変数 $V f(1) \sim V f(16)$ が得られる。

ちなみに、本実施形態では変数名を節約するため、正規化する前と正規化した後で、同じ $V f(n)$ という変数名を使用することとする。

【 0 0 2 8 】

後輪側の正規化手段 1 5 r も、前輪側の正規化手段 1 5 f と同様の正規化処理を行う（重複を避けるために説明を簡略化する）。即ち、正規化手段 1 5 r は、後輪側の平均車輪速 AV_r を次の式 3 で演算する。

$$\begin{aligned} AV_r &= \sum V_r(m) / 30 \\ &= (V_r(1) + V_r(2) + \dots + V_r(30)) / 30 \quad \dots \quad (\text{式 3}) \end{aligned}$$

【0029】

また、後輪側の正規化手段 1 5 r は、平均車輪速 AV_r を用いて、次の式 4 により、正規化を行う。

$$V_r(m) = V_r(m) - AV_r \quad \dots \quad (\text{式 4})$$

【0030】

なお、処理カウンタ m は 1 ～ 30 までの正の整数であるので、正規化手段 1 5 r は、処理カウンタ m を 1 から 1 ずつインクリメントして終値 M の 30 になるまで 30 回、式 4 を実行する。これにより、正規化した配列変数 $V_r(1) \sim V_r(30)$ が得られる。

【0031】

相互相関関数演算手段 1 6 は、フーリエ変換の一種である相互相関関数を演算（実行）する手段である。つまり、この相互相関関数演算手段 1 6 は、前記した 1 50 ミリ秒の間に前輪 W_f に現われる路面バンプ等による車輪速の変動パターンと同じ変動パターンが、290 ミリ秒の間に後輪 W_r にどのように（どの時点で）現われるのかを判断するための処理を行う手段である。このため、相互相関関数演算手段 1 6 は、正規化手段 1 5（1 5 f, 1 5 r）から一括して正規化した配列変数 $V_f(n)$, $V_r(m)$ を取得して、次の式 5 ～ 式 19 に示すように畳み込み積分を行う（式 8 ～ 式 18 は省略）。

$$\begin{aligned} S(1) &= V_f(1) \cdot V_r(1) + V_f(2) \cdot V_r(2) + \dots + V_f(16) \cdot V_r(16) \\ &\dots \quad (\text{式 5}) \end{aligned}$$

$$\begin{aligned} S(2) &= V_f(1) \cdot V_r(2) + V_f(2) \cdot V_r(3) + \dots + V_f(16) \cdot V_r(17) \\ &\dots \quad (\text{式 6}) \end{aligned}$$

$$\begin{aligned} S(3) &= V_f(1) \cdot V_r(3) + V_f(2) \cdot V_r(4) + \dots + V_f(16) \cdot V_r(18) \\ &\dots \quad (\text{式 7}) \end{aligned}$$

・ ・ (中略) ・ ・

$$S(15) = V_f(1) \cdot V_r(15) + V_f(2) \cdot V_r(16) + \dots + V_f(16) \cdot V_r(30) \quad \dots (式 19)$$

【 0 0 3 2 】

ここで、 $S(1) \sim S(15)$ は $S(j)$ として表現されるが、この $S(j)$ は、相互相関関数の演算（畳み込み積分）の結果を 15 個分 ($j = 1 \sim 15$) 格納する配列変数である。なお、 j は、データのアドレスを指定するインデックスである。

【 0 0 3 3 】

ところで、相互相関関数の演算が完了して配列変数 $S(j)$ に結果のデータが書き込まれると、新たな検出値 $V(V_f, V_r)$ を配列変数 $V_f(n)$, $V_r(m)$ として記憶しても車体速 V_v の測定に支障は生じない。このため、相互相関関数演算手段 16 は、相互相関関数の演算が完了すると、処理完了報告（図示外）をバッファコントローラ 13 (13_f , 13_r) に行うものとする。バッファコントローラ 13 は処理完了報告を受信すると、処理カウンタ n , m を 0 にし、新たな検出値 $V(V_f, V_r)$ を、前記したように配列変数 $V_f(n)$, $V_r(m)$ として新たにデータバッファ 14 に記憶するものとする。このようにすることで、変数名を節約することができる。即ち、メモリの使用量を節約することができ、例えばより多くのデータを取り扱えるようになる。

【 0 0 3 4 】

ちなみに、データバッファ 14 から配列変数 $V_f(n)$, $V_r(m)$ とは異なる変数名をつけて配列変数 $V_f(n)$, $V_r(m)$ を読み出すとすれば、配列変数 $V_f(n)$, $V_r(m)$ の読出完了後、直ちに新たな検出値 V_f , V_r を同じ変数名の配列変数 $V_f(n)$, $V_r(m)$ として記憶することもできる。このようにすると、処理時間を短縮化することが可能になる。

【 0 0 3 5 】

次に、最大値抽出手段 17 は、配列変数 $S(j)$ のうち、最大値を抽出する関数を実行する手段である。つまり、前記した畳み込み積分の結果が割り当てられている配列変数 $S(j)$ から、次の式 20 により最大値を抽出する。

$$S_{sim} = \max | S(1), S(2), S(3), \dots, S(15) | \quad \dots \quad (\text{式 } 20)$$

【0036】

車体速演算手段18は、前記した配列変数が最大値となるインデックスjの値から時間差 Δt を決定する処理、及び別に記憶している車両Cの前輪Wfと後輪Wfのホイールベース間距離（基準長さ）WBの値とから、次の式21、式22により車体速Vvを演算する処理を行う手段である。

$$\Delta t [\text{秒}] = 10 [\text{ミリ秒}] / 1000 [\text{ミリ秒/秒}] \times (j - 1) \quad \dots \quad (\text{式 } 21)$$

$$Vv [\text{km/hr}] = WB [\text{m}] / \Delta t [\text{秒}] \times 3600 [\text{秒/hr}] / 1000 [\text{m/km}] \quad \dots \quad (\text{式 } 22)$$

【0037】

なお、時間差 Δt は、請求項の「一致したパターンの時間差」に相当する。また、式21の10という値は、各検出値Vf、Vrのサンプリング間隔である。また、インデックスjから1を引くのは、区間数を求めるためである。

【0038】

平均車体速演算手段19は、図示しないFIFO（First In First Out）を有し、車体速演算手段18で演算された車体速Vvから平均車体速AVvを演算する処理を行う手段である。ここで、FIFOは、先入れ先出しを行うメモリである。FIFOには、車体速演算手段18で演算されたK個の車体速Vvが配列変数Vv(k)として格納されるものとし、新たな車体速Vvが車体速演算手段18からFIFOに配列変数Vv(1)として格納される際、最も古い配列変数Vv(K)は消去する。そして、他の配列変数Vv(k)は、そのインデックスkを1だけインクリメントされる。つまり、配列変数Vv(1)は配列変数Vv(2)に、配列変数Vv(K-1)は配列変数Vv(K)に、というようにインデックスkがインクリメントされ、過去の一定時間にわたる車体速Vvが順に更新される。なお、インデックスkとその終値Kの関係は、 $1 \leq k \leq K$ 、となる（但し $K > 1$ ）。ちなみに、終値Kは例えば5である。

【0039】

そして、平均車体速演算手段19は、FIFOから配列変数Vv(1)～Vv

(5) を読み出して、次の式 2 3 により平均車体速 AV_v を演算する（つまり車体速を測定する）。

$$AV_v = \sum V_v(k) / K$$

$$= (V_v(1) + V_v(2) + \dots + V_v(K)) / K \quad \dots \quad (\text{式 2 3})$$

【0 0 4 0】

ちなみに、インデックス k の終値 K が 5 であるとする、処理カウンタ m の終値 M が 3 0 であり、各検出値 V (V_f , V_r) のデータ間隔が 1 0 ミリ秒であることから平均車体速 AV_v は、1. 5 秒間 (= 3 0 × 1 0 ミリ秒 × 5) における車両 C の平均した車体速となる。なお、前記したとおり、バッファコントローラ 1 3 (13_f , 13_r) は、検出値 V (V_f , V_r) を常に 1 0 ミリ秒間隔でデジタルフィルタ 1 2 から取得し、データバッファ 1 4 (14_f , 14_r) に記憶するものとする。

【0 0 4 1】

このように、平均車体速演算手段 1 9 を、F I F O を有する構成とすることで、平均車体速演算手段 1 9 は、車体速演算手段 1 8 が車体速 V_v を 1 つ演算するごとに、平均車体速 AV_v を 1 つ演算（測定）することができる。なお、変数名を節約するため、車体速演算手段 1 8 が演算した車体速 V_v と平均車体速演算手段 1 9 が演算した平均車体速 AV_v を同じ変数名 V_v にしてもよい。

このように演算（測定）された平均車体速 AV_v は、各種制御に使用される。

【0 0 4 2】

ちなみに、本実施形態において、デジタルフィルタ 1 2、バッファコントローラ 1 3、データバッファ 1 4、正規化手段 1 5 で行われる処理は、請求項の「タイヤ固有の影響を除去」、及び「特徴抽出」に相当する。また、相関関数演算手段 1 6、最大値抽出手段 1 7、車体速演算手段 1 8 で行われる処理は、請求項の「パターンマッチング」、及び「車体速を演算」に相当する。

【0 0 4 3】

《車体速測定装置の動作》

次に、本実施形態の車体速測定装置の動作を、図 1 ～図 7 を参照して説明する。図 4 は、車体速測定の様子を模式的に示す図であり、(a) は車両が a 地点及

び b 地点を含む道路を b 地点側へと走行する様子を模式的に示し、(b) はその際における車輪速の検出値の変化を時系列で示し、(c) は (b) の検出値をデジタルフィルタで処理した後の検出値の変化を時系列で示す。図 5 は、車体速を測定する処理のうち、検出値をデータバッファに記憶する部分に関するフローチャートである。図 6 は、車体速を測定する処理のうち、データバッファに記憶された検出値から平均車体速を演算する部分に関するフローチャートである。図 7 は、(a) が正規化処理後の配列変数 $V_f(n)$ を模式的に示し、(b) が正規化処理後の配列変数 $V_f(m)$ を模式的に示す。

【 0 0 4 4 】

〔タイヤのユニフォーミティの崩れによる変動の除去〕

図 4 (a) に示すように、車両 C がある車速で a 地点、b 地点を含む道路を走行する。車両 C が走行すると車輪速センサ V_S (V_{Sf} , V_{Sr}) から入出力インタフェース 11 を介して車輪速の検出値 V (V_f , V_r) が車体速測定装置 1 に入力される。前記したとおり前輪 W_f 、後輪 W_r のタイヤにはユニフォーミティの崩れが存在するのでこれによる周期の大きな変動と、路面バンプ等による周期の小さな変動が車輪速センサ V_S で検出される検出値 V_f , V_r に重畳されている (図 4 (b) 参照)。つまり、見かけ上車両 C が一定速で走行していても、タイヤのユニフォーミティの崩れと路面バンプ等の存在による影響で検出値 V_f , V_r は変動する。本実施形態では、車体速を路面バンプ等による車輪速変動から測定するので、デジタルフィルタ 12 で処理してタイヤのユニフォーミティの崩れによる変動を検出値 V_f , V_r から除去する。

なお、図 4 (b) の上図は前輪側についてのものであり、下図は後輪側についてのものであり、車両 C のホイールベース間距離 WB 及び車体速 V_v に対応した位相のズレがある。つまり、前輪 W_f が a 地点 (b 地点) を通過した後、後輪 W_r が a 地点 (b 地点) を通過する。この点は、次の図 4 (c) でも同じである。ちなみに、図 4 (c) は、図 4 (b) よりも縦軸方向を強調して記載している。

【 0 0 4 5 】

デジタルフィルタ 12 で処理すると、図 4 (c) に示すように検出値 V_f , V_r からタイヤのユニフォーミティの崩れによる変動が除去される。これにより、

絶対的な速度である車体速をより正確に測定（演算）できるようになる。

【0046】

〔検出値のデータバッファへの記憶〕

デジタルフィルタ12により、タイヤのユニフォーミティの崩れによる変動が除去された検出値 V_f 、 V_r は、バッファコントローラ13により10ミリ秒間隔で取得され、データバッファ14に配列変数 $V_f(n)$ 、 $V_r(m)$ として記憶する処理が次に行われる。

【0047】

即ち（主として図5参照）、最初は、処理カウンタ n 、 m が0になっているので、処理カウンタ n 、 m をインクリメントして、検出値 $V(V_f, V_r)$ をデータバッファ14に記憶する。具体的には、前輪側についての処理カウンタ n をインクリメントする（S11）。そして、前輪側についてデジタルフィルタ12fが処理した検出値 V_f を10ミリ秒間隔で取得して、データバッファ14rに配列変数 $V_f(n)$ として記憶する（S12）。後輪側でも、ステップS11に相当する処理をステップS13で、ステップS12に相当する処理をステップS14で行う。

【0048】

データバッファ14rに検出値 V_r を配列変数 $V_r(m)$ として記憶すると、バッファコントローラ13が、処理カウンタ n の値が終値 N になっているか否かを判断する（S15）。終値 N になっていない場合（no）は、再びステップS11に戻り処理を続行する。これにより、データバッファ14に順次検出値 V_f 、 V_r が配列変数 $V_f(n)$ 、 $V_r(m)$ として記憶される。

【0049】

一方、ステップS15で処理カウンタ n が終値 N になっている場合（yes）、つまり本実施形態でいえば処理カウンタ n が16になった場合は、処理カウンタ m が終値 M になっているか否かを判断する（S16）。

【0050】

ステップS16で、処理カウンタ m が終値 M になっていない場合は、ステップS13に戻り、処理カウンタ m をインクリメントした後、後輪側のデジタルフィ

ルタ 1 2 r が処理した検出値 V_r をデータバッファ 1 4 r に配列変数 $V_r(m)$ として記憶する (S 1 4)。この後は、ステップ S 1 5 を経由してステップ S 1 6 の処理を繰り返す。これにより、後輪側のデータバッファ 1 4 r にだけ、順次検出値 V_r が配列変数 $V_r(m)$ として記憶される。

【 0 0 5 1 】

一方、ステップ S 1 6 で処理カウンタ m が終値 M になっている場合 (yes)、つまり本実施形態でいえば処理カウンタ m が 3 0 になった場合は、相互相関関数演算手段 1 6 からの処理完了報告の有無を判断する (S 1 7)。処理完了報告がない場合 (no) は、処理完了報告があるまで処理を待つ。処理完了報告が相互相関関数演算手段 1 6 からバッファコントローラ 1 3 にあった場合 (yes) は、バッファコントローラ 1 3 は、処理カウンタ n, m を 0 にクリアした後 (S 1 8)、Return に移行する (処理を継続する)。

【 0 0 5 2 】

これにより、1 0 ミリ秒間隔ごとに、検出値 V_f が 1 6 個分、配列変数 $V_f(n)$ として記憶され、検出値 V_r が 3 0 個分、配列変数 $V_r(m)$ として記憶され、次の処理である平均車体速の演算処理の前準備が整う。

【 0 0 5 3 】

〔平均車体速の演算〕

データバッファ 1 4 f, 1 4 r に配列変数 $V_f(n), V_r(m)$ が所定個数記憶されると、図 6 のフローチャートに示すように、データバッファ 1 4 から配列変数 $V_f(n), V_r(m)$ を全て読み出す (S 2 1)。そして、前輪側及び後輪側について既に説明した手順により正規化を行う (S 2 2, S 2 3)。この際の演算において使用されるのは、式 1 ~ 式 4 である。正規化が完了すると、図 7 (a), (b) のようなグラフで配列変数 $V_f(n), V_r(m)$ が模式的に示される。なお、既に説明したように、正規化する前と正規化する後とで、同じ変数名を使用して、メモリを節約している。

【 0 0 5 4 】

ステップ S 2 2, S 2 3 で正規化が完了すると、既に説明した式 5 ~ 式 1 9 を使用して相互相関関数を演算する (S 2 4)。なお、式 5 ~ 式 1 9 は、次のよう

に1つの式24にまとめて、繰り返し部分を省略化することができる。

$$S(j) = V_f(1) \cdot V_r(1+j) + V_f(2) \cdot V_r(2+j) + \dots + V_f(16) \cdot V_r(16+j) \quad \dots \quad (\text{式 } 24)$$

【0055】

ステップS24で、相互相関関数を演算して配列変数 $S(j)$ に格納すると、配列変数 $V_f(n)$ 、 $V_r(m)$ に新しいデータを書き込むことができるようになる。このため、ステップS25で、相互相関関数演算手段16が処理完了報告をバッファコントローラ13に出力する。これにより、新たな検出値 V_f 、 V_r を配列変数 $V_f(n)$ 、 $V_r(m)$ に格納してデータバッファ14に記憶することができるようになる(図5のステップS17参照)。

【0056】

ステップS26では、式20により、相互相関関数の演算結果を格納した配列変数 $S(j)$ から最大値を抽出する(S26)。そして、その最大値となる $S(j)$ のインデックス j を特定し、このインデックスを式21に代入して時間差 Δt を決定する。続けて、式22に決定した時間差 Δt と予め記憶しているホイールベース間距離 WB を代入して、車体速 V_v を演算する(S27)。ちなみに、ステップS24の相互相関関数の演算、及びステップS26の最大値の抽出は、図7(a)のグラフに図7(b)のグラフをどの様にずらせば両グラフが重なり合うのかを試行(パターンマッチング)することに相当し、ステップS27の時間差 Δt の決定は、重なり合う場所における両グラフの位相差を決定するものである。

【0057】

位相差の決定を、図7(a)、(b)と式5~19を用いて補足説明する。

前輪側と後輪側とで位相が揃わない場合(パターンの異なる場合)の式5では、例えば「 $V_f(2)$ と $V_r(2)$ の積」、「 $V_f(3)$ と $V_r(3)$ の積」は負の値になり、例えば「 $V_f(16)$ と $V_r(16)$ の積」は正の値になる。従って、和の $S(1)$ は、正の値と負の値を足し合わせて演算されることになる。

位相が揃わない場合の式6等も同様であり、和の $S(2)$ 等は、正の値と負の値を足し合わせて演算されることになる(図7(a),(b)参照)。

ところが、位相が揃う場合（パターンが一致する場合）の式19では、「 $V_f(1)$ と $V_r(15)$ の積」～「 $V_f(16)$ と $V_r(30)$ の積」の全てが正の値になるので、和の $S(15)$ も、 $S(j)$ の中で最も大きな値になる（ j は1～15）。

このことから、最大値となる $S(j)$ のインデックス j を見つけ出せば、そのインデックス j とサンプリング間隔（ここでは10ミリ秒）から位相差がどれだけの時間あるのかが判る。

【0058】

次に、ステップS27の処理を、具体的な数字を用いて説明する。

仮に、配列変数 $S(15)$ が最大値であったとすると（S26）、時間差 Δt は140ミリ秒（ $= (15-1) \cdot 10 \text{ミリ秒} = 0.14 \text{秒}$ ）になる。ここで、ホイールベース間距離 WB が2.83mとすると車体速 V_v は、式22により次のように求められる。

$$\begin{aligned} V_v &= WB / \Delta t \times 3600 / 1000 \quad \dots \quad (\text{式22}) \\ &= 2.83 / 0.14 \times 3.6 \\ &= 73 [\text{km/hr}] \end{aligned}$$

【0059】

このステップS27で演算された車体速 V_v は、S28で平均車体速 AV_v を演算するのに使用される。即ち、ステップS28では、車体速 V_v が演算されると、これを配列変数 $V_v(k)$ として先入れ先出しメモリのFIFOに記憶する。そして、平均車体速 AV_v を式23に基づいて演算する（移動平均の演算）。演算後はReturnに移行して処理を継続する。これにより、ステップS21～ステップS28が順次繰り返され、平均車体速 AV_v が演算される。なお、ステップS28において、最初のうちはFIFOにデータが全て記憶されていないことから、ステップS28（平均車体速演算手段19）では、記憶したデータの分だけの平均車体速 AV_v を演算するものとする。ちなみに、FIFOにデータが全て記憶された後は、データバッファ14に検出値 V_f 、 V_r が配列変数 $V_f(n)$ 、 $V_r(m)$ として所定個数書き込まれると、これを全て読み出して（S21）、ステップS22～S28の処理を実行して直ちに平均車体速 AV_v を演

算する。

【0060】

なお、フローチャートのステップS11～S23までが、請求項の「特徴抽出」に相当し、フローチャートのステップS24，S26が請求項の「パターンマッチング」に相当するといえる。

【0061】

このようにして、本実施形態の車体速測定装置1は、車輪速センサVSにより検出される車輪速の検出値Vの変動（即ちタイヤを介して入力される路面との振動）に基づいて平均車体速を演算（測定）する。このような測定によれば、タイヤの径が変化しても、車体速をより正しく測定することができる。この点、従来の文字通りの車輪速に基づいて車体速を測定するのとは異なる。

【0062】

なお、本実施形態で測定（演算）した車体速V_v及び平均車体速AV_vは、原理上タイヤのサイズ、空気圧の多少等に影響されない速度である。従って、測定値は、各種の制御や車両状況の検出等に利用することができる。例えば、タイヤの空気圧の漏れ判定（パンク検出）では、4輪ごとに設置した車輪速センサから車輪速の検出値を監視し、この車輪速の検出値と本実施形態の車体速とを比較して、車体速よりも早い車輪速にかかる車輪（タイヤ）を空気漏れと判定することができる。また、カーナビゲーションでは、長いトンネル内を走行する際の車両位置の検出に、本実施形態の車体速を使用することができる。また、ヨーレートの算出等にも本実施形態の車体速を使用することができる。また、得られる車体速の精度を高いものとすることも可能である。

【0063】

なお、以上説明した本発明は、前記実施形態に限定されことなく幅広く変形実施することができる。

例えば、既に説明したように、後輪側の変動パターンと同じパターンが、前輪側でいつ出現したかをパターンマッチングにより検出して車体速を測定することとしてもよい。また、平均車体速を演算しない構成としてもよい。また、最大値となるS(j)に閾値を定め、最大値となるS(j)であっても、ある値（閾値

) を超えなければ、車体速 V_v を演算しないようにしてもよい。

【0064】

処理カウンタの終値は一例であり、必ずしも前記した実施形態の値に限定されることはない。また、検出値のサンプリング間隔を10ミリ秒ごととして説明したが、これを車速が早くなると短くなるようにしてもよい。また、処理カウンタの終値も検出値のサンプリング間隔や車速に応じて変動するようにしてもよい。

また、タイヤのユニフォーミティの崩れによる変動を、デジタルフィルタでソフトウェア的に除去したが、ハードウェア的に除去するようにしてもよい。また、バンドパスフィルタを用いた例を説明したが、ローパスフィルタやハイパスフィルタ等を用いるようにしてもよい。車体速測定装置における処理を、ハードウェア的に行うようにしてもよい。また、図1に示すように車輪速センサ V_S を片側の前輪 W_f 、片側の後輪 W_r に有することとしたが、両側に有することとしてもよい。また、配置が斜になるようにしてもよい。

【0065】

また、振動検出センサとして、車輪速センサを例に説明したが、路面と車両（タイヤ）との間で発生する振動を検出することのできるものであれば、変位計、重量計、Gセンサ等、様々なセンサを適用することができる。例えば、前輪側、後輪側のサスペンションにこれらセンサを設置するようにし、この検出値により車体速を測定（演算）するようにしてもよい。

【0066】

また、相互相関関数によるパターンマッチングは一例であり、本発明がこれに限定されることはない。

【0067】

【発明の効果】

請求項1に記載の発明によれば、原理的にタイヤ径の変化に影響されない車体速の測定が可能になる。また、請求項2に記載の発明によれば、車両が備えている車輪速センサの検出値をそのまま利用することができるので、例えばコスト的にも機器のレイアウト的にも都合がよい。

【図面の簡単な説明】

【図 1】 本発明にかかる実施形態の車体速測定装置を搭載する車両のシステム構成図である。

【図 2】 車体速測定装置の要部をなす車体速測定装置のブロック構成図である。

【図 3】 車輪速の検出値の変動を説明する図である。

【図 4】 車体速測定の様子を模式的に示す図であり、(a)は車両が a 地点及び b 地点を含む道路を b 地点側へと走行する様子を模式的に示し、(b)はその際における車輪速の検出値の変化を時系列で示し、(c)は(b)の検出値をデジタルフィルタで処理した後の検出値の変化を時系列で示す。

【図 5】 車体速を測定する処理のうち、検出値をデータバッファに記憶する部分に関するフローチャートである。

【図 6】 車体速を測定する処理のうち、データバッファに記憶された検出値から平均車体速を演算する部分に関するフローチャートである。

【図 7】 (a)が正規化処理後の配列変数 $V_f(n)$ を模式的に示し、(b)が正規化処理後の配列変数 $V_f(m)$ を模式的に示す。

【符号の説明】

- 1 … 車体速測定装置
- 1 1 … 入出力インタフェース
- 1 2 … デジタルフィルタ
- 1 3 … バッファコントローラ
- 1 4 … データバッファ
- 1 5 … 正規化手段
- 1 6 … 相互相関関数演算手段
- 1 7 … 最大値抽出手段
- 1 8 … 車体速演算手段
- C … 車両
- V S … 車輪速センサ（振動検出センサ）
- V, V_f , V_r … 検出値
- $V_f(n)$, $V_r(m)$, $S(j)$ … 配列変数

V_v ... 車体速

$A V_v$... 平均車体速

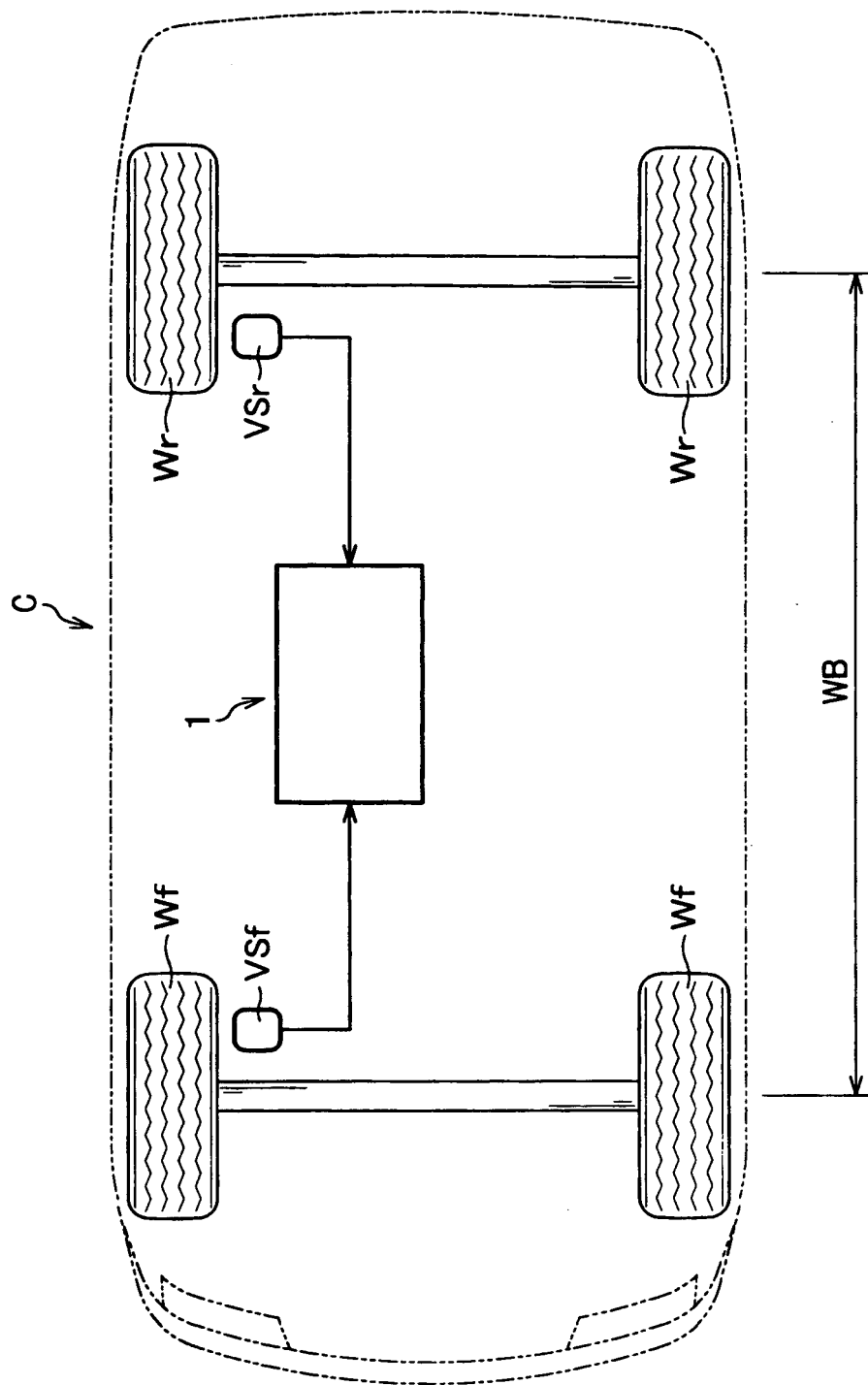
Δt ... 時間差

WB ... ホイールベース間距離（基準長さ）

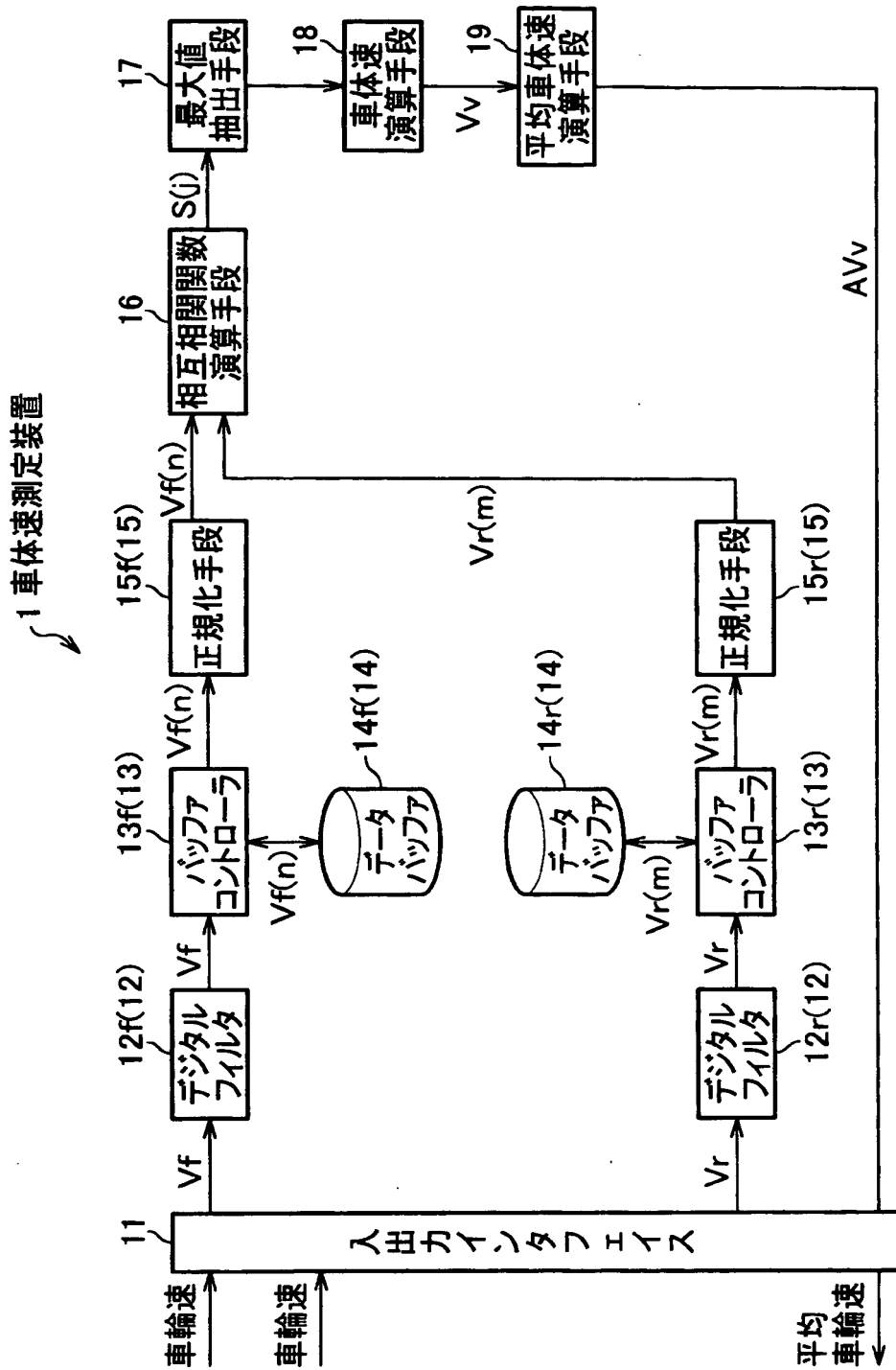
【書類名】

図面

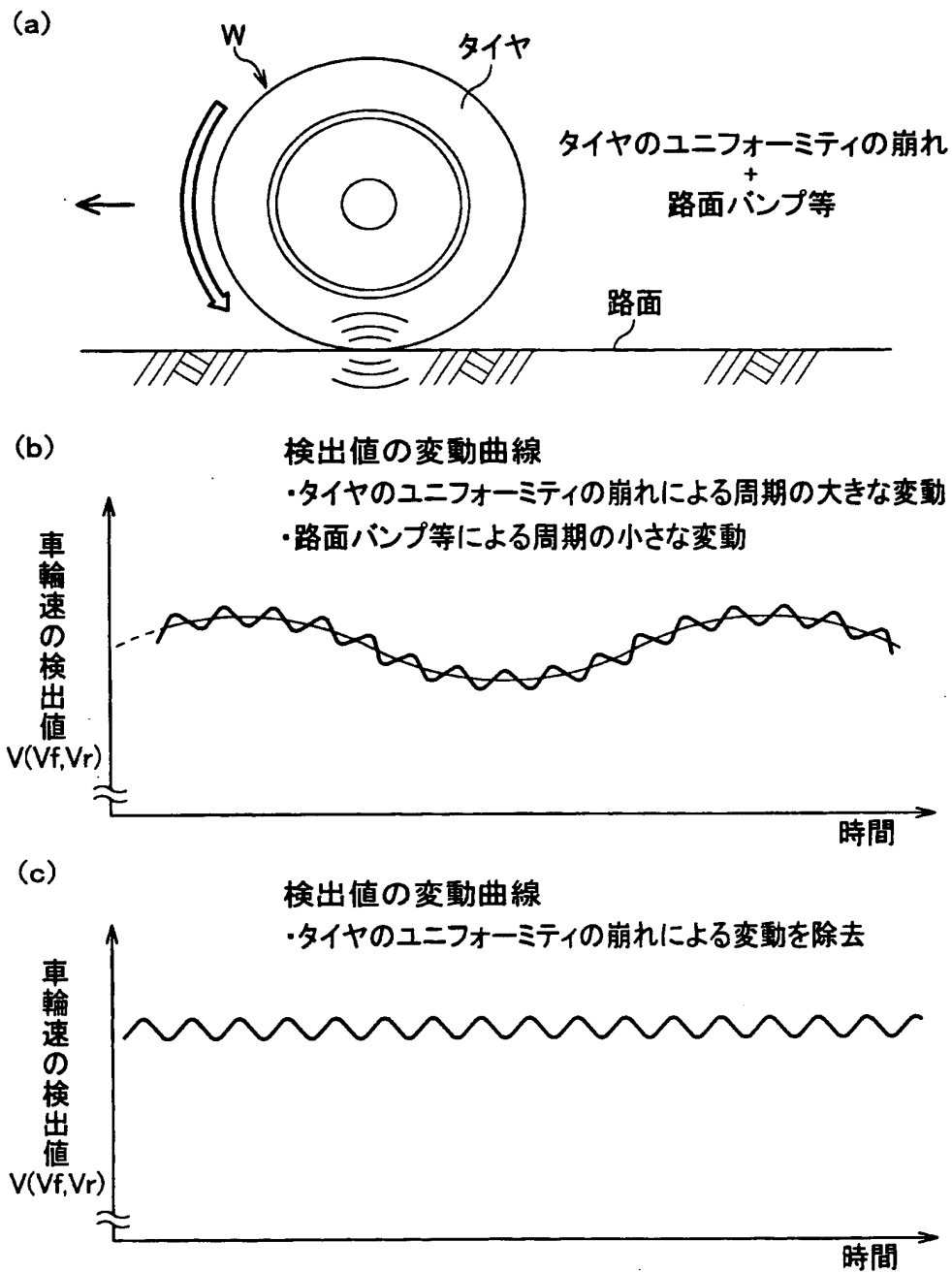
【図 1】



【図 2】

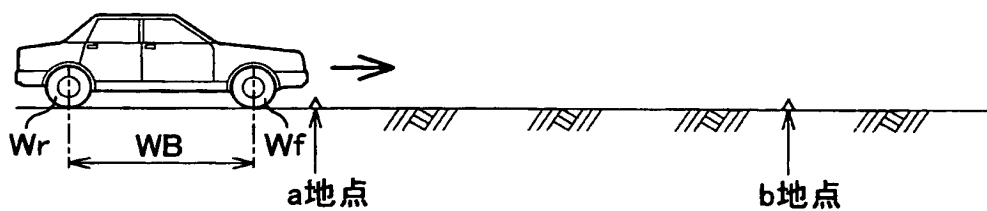


【図 3】

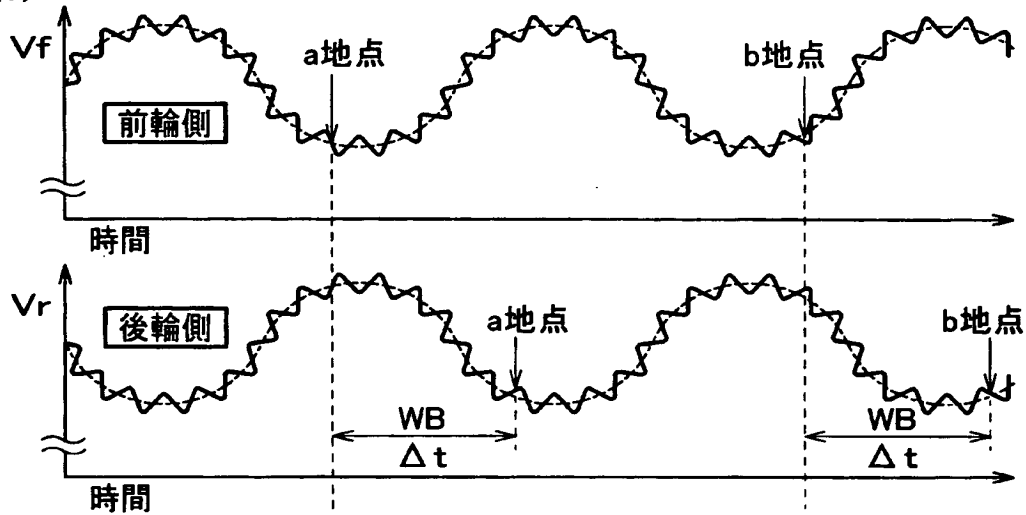


【図 4】

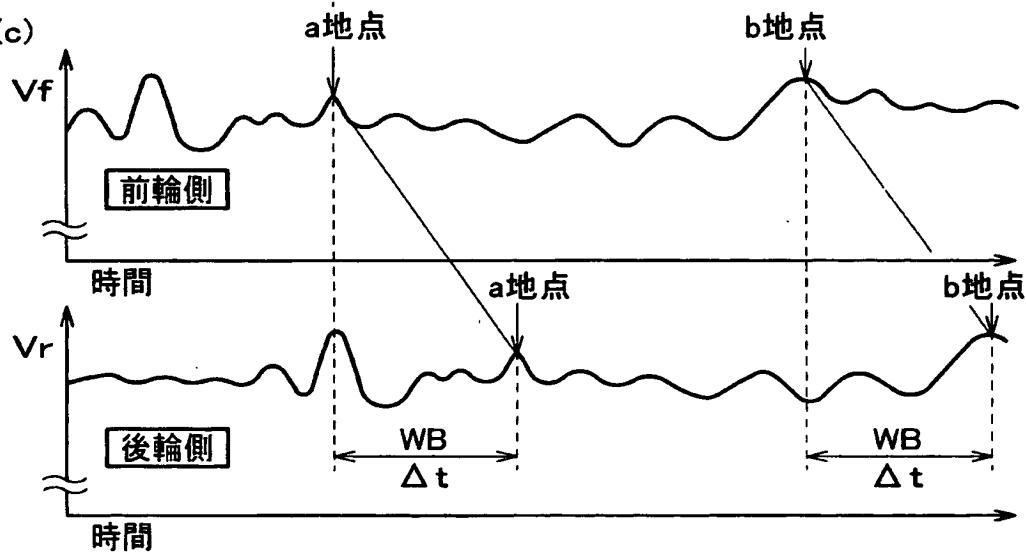
(a)



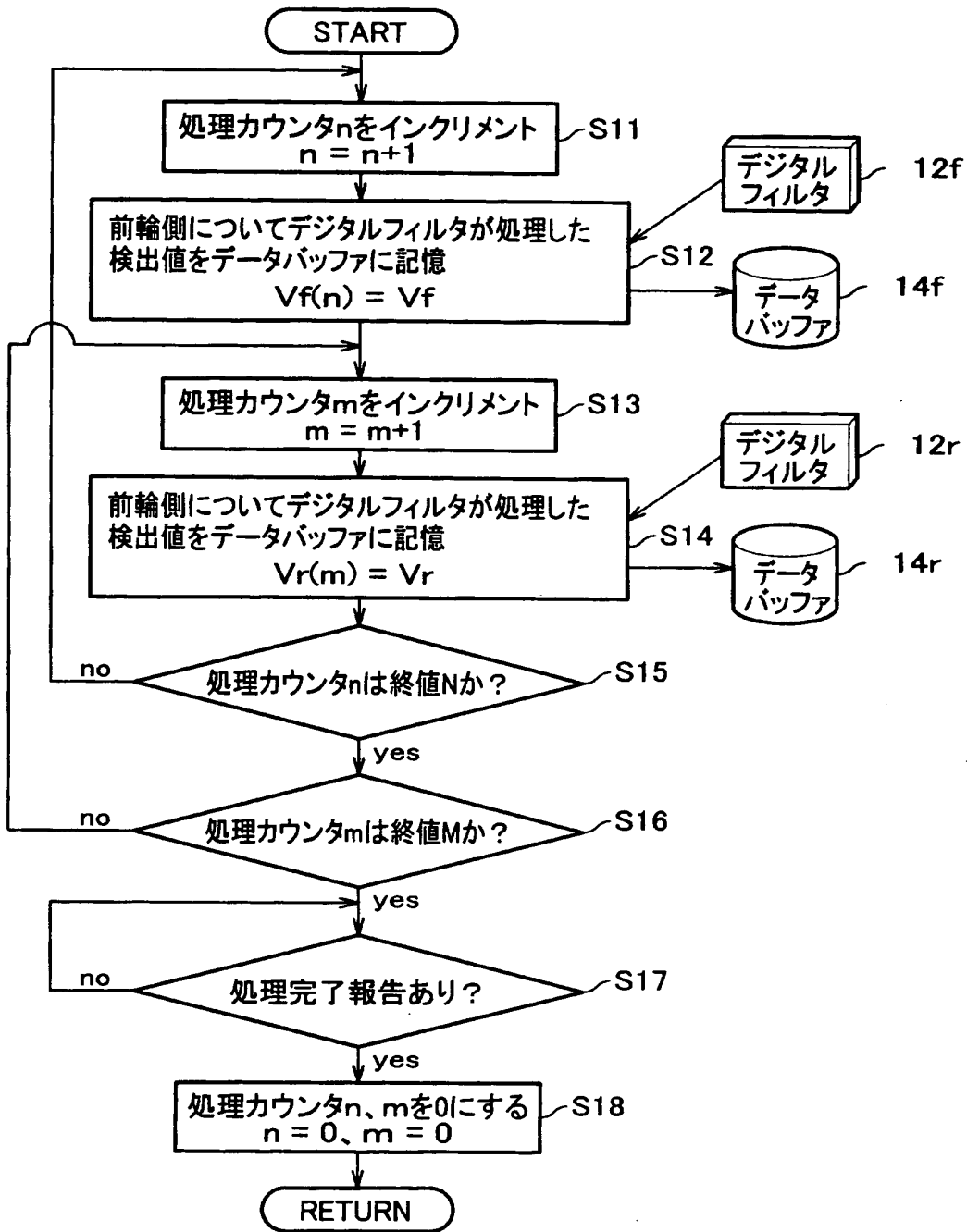
(b)



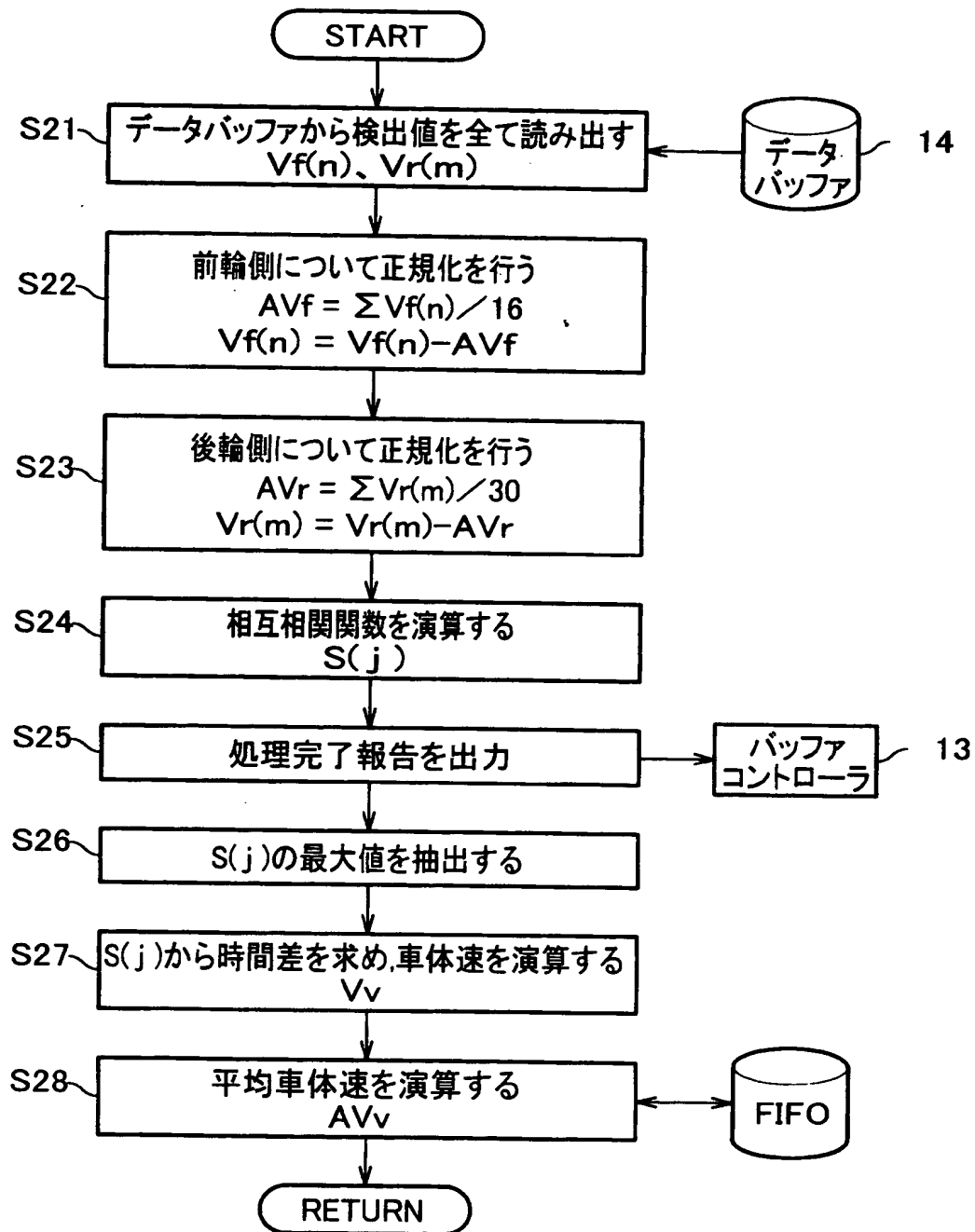
(c)



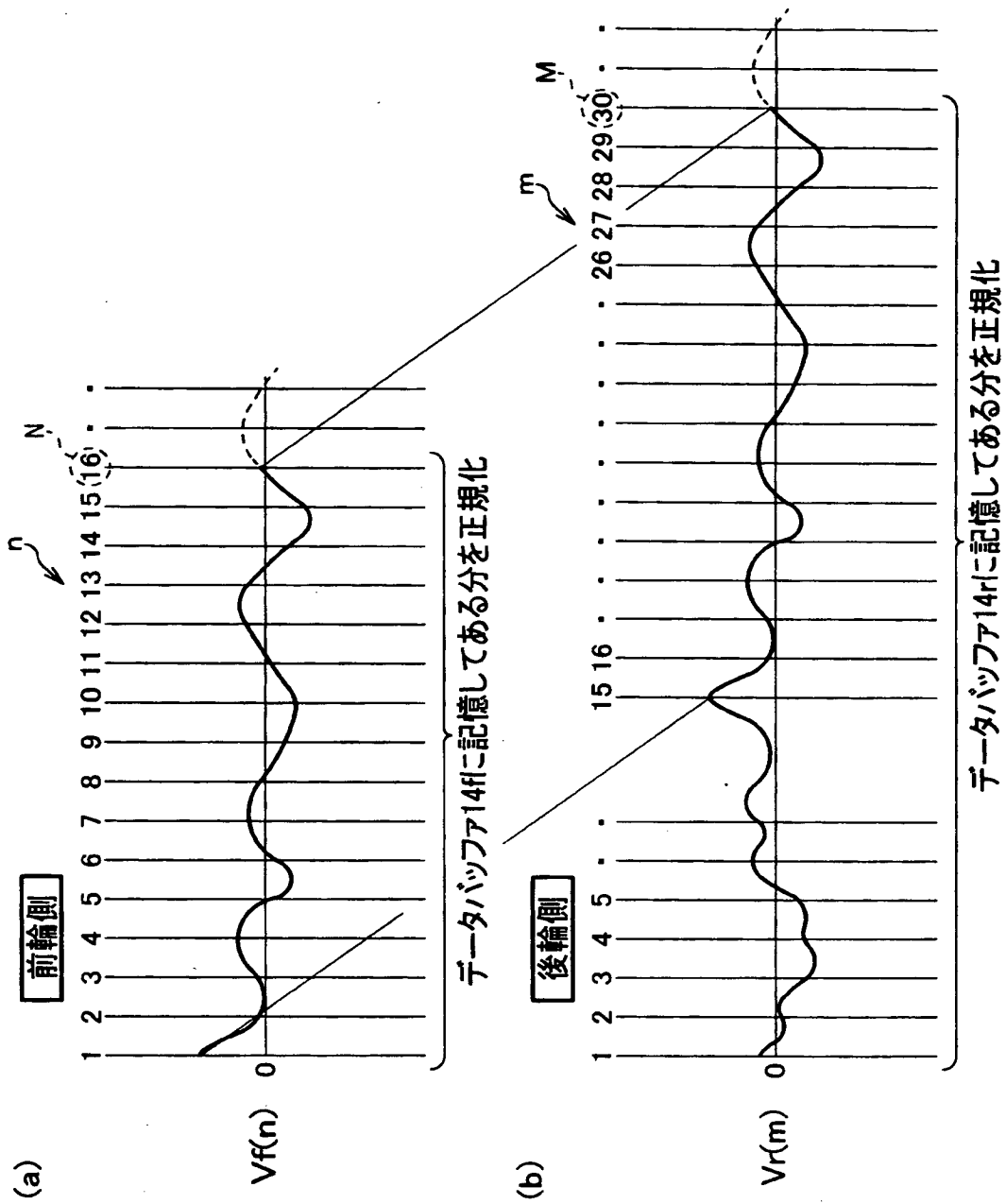
【図 5】



【図 6】



【図 7】



【書類名】 要約書

【要約】

【課題】 タイヤの径の変化の影響が少ない車体速の測定を可能とする車体速測定装置を提供すること。

【解決手段】 少なくとも入力部 1 1 と処理部 1 2 …を有し、タイヤを介して入力される路面との振動を検知する前輪側及び後輪側の振動検出センサから検出値 V_f , V_r をそれぞれ入力し、この入力した検出値の変化のパターンに基づいて車両の車体速 V_v を測定する。このため、入力部 1 1 から検出値 V_f , V_r を入力すると、処理部 1 2 …にて、前輪側及び後輪側について検出値 V_f , V_r の変化のパターンをタイヤ固有の影響を除去してそれぞれ特徴抽出する。この特徴抽出した検出値 $V_f(n)$, $V_r(m)$ の変化のパターンを前輪側と後輪側とでパターンマッチングし、一致したパターンの時間差 Δt を求め、このように求めた時間差 Δt と予め記憶しているホイールベース間距離 WB とから車体速 V_v を演算するようにした。

【選択図】 図 2

出 願 人 履 歴 情 報

識別番号 [0 0 0 0 0 5 3 2 6]

1. 変更年月日	1 9 9 0 年 9 月 6 日
[変更理由]	新規登録
住 所	東京都港区南青山二丁目 1 番 1 号
氏 名	本田技研工業株式会社